

I HEREBY CERTIFY THAT THIS CORRESPONDENCE IS BEING DEPOSITED WITH THE UNITED STATES POSTAL SERVICE AS EXPRESS MAIL IN AN ENVELOPE ADDRESSED TO: ASSISTANT COMMISSIONER FOR PATENTS, WASHINGTON, D.C. 20231. THE APPLICANT AND/OR ATTORNEY REQUESTS THE DATE OF DEPOSIT AS THE FILING DATE.

Express Mail No: EK140252742US

Date of Deposit: 12/21/01

Name of Person Making Deposit: K. C. INO-MARS

Signature: *Ken Ling-Mars* 12/21/01

APPLICATION
FOR
UNITED STATES LETTERS PATENT

APPLICANT:

Darren L Anand et al.

FOR:

Flexible Multibanking Interface for Embedded Memory Applications

DOCKET:

FIS9-2001-0253-US1

INTERNATIONAL BUSINESS MACHINES CORPORATION
ARMONK, NEW YORK 10504

Flexible Multibanking Interface for Embedded Memory Applications

Background of the Invention

The present invention relates generally to an interface for embedded DRAM.

5 In most applications, the DRAM memory is intended to support the operation of one or more logic devices (e.g., microprocessors). In conventional DRAM applications, a dedicated chip(s) containing the DRAM memory was typically employed separate from the chip containing the microprocessor(s) or other logic devices. The data/control transmission between the DRAM and the
10 logic device would typically be managed through a bus architecture on the circuit board containing the various chips. In such conventional configurations, the selection of DRAM architecture and interface was largely predetermined by the DRAM manufacturer and/or industry standards for the particular type of DRAM memory.

15 Where logic device manufacturers (e.g., ASICs manufacturers) had desired memory on the logic chip, this was generally accomplished using static random access memory (SRAM). Unfortunately, SRAM consumes a significant amount of chip area relative to the memory capacity provided (i.e., a single SRAM data cell usually requires on the order of six transistors). Thus, more
20 recently, logic device manufacturers have moved toward the use of eDRAM for on-chip memory in order to conserve chip area and/or increase performance through more on-chip memory capacity. The motivation to replace SRAM implementations with eDRAM places additional pressure on macro performance and random cycle time.

The desire to embed DRAM into logic chips has resulted in the need for development of DRAM macro designs which are amenable to logic design environments. The logic or ASIC environment presents many difficult challenges that DRAMs have historically been sensitive to, such as wide voltage and
5 temperature ranges and uncertainties in surrounding noise conditions. There is a need for DRAM macro designs having performance characteristics which can be varied by the logic designer without significant restructuring of the DRAM macro. There is also a desire for DRAM macro designs having improved bus utilization and speed.

10 Summary of the Invention

The invention provides a synchronous multibank interface on a growable embedded DRAM. The flexible multibank interface of the invention can be grown without redesign and without change of appearance/behavior to the customer. The interface is preferably characterized by the presence of bank
15 select inputs (pins) which permit selection of one or more banks of the macro. The banks preferably each have respective support circuitry including row decode circuitry and respective limited repair redundancy. The interface of the invention has advantages over typical synchronous DRAM interfaces in that operations to each bank are independent. Also, activation and pre-charge
20 operations can both be completely hidden during reads and writes. The interface of the invention may be used to support multiple bank configurations with any number of banks (i.e., one or more banks).

In one aspect, the invention encompasses a multibank DRAM macro comprising:

- 25 (a) a plurality of DRAM memory banks, each bank respectively comprising:
 - (i) an array of DRAM memory cells,

- 5
- (ii) bitlines and wordlines, respectively defining columns and rows of the array, and
 - (iii) a row address decoder circuit,
 - (iv) a column address decoder circuit,
 - (v) spare rows and columns for redundancy,
- (b) a bank select input for each respective bank, each bank input controlling operation of its respective bank, and
- (c) a data path receiver / driver shared by at least two banks.

10 The macro preferably further includes a master select input capable of latching all the bank select inputs and row address inputs. The macro further includes a shared data path for all banks.

In another aspect, the invention encompasses a methods for operating the eDRAM macro of the invention.

15 These and other aspects of the invention are described in further detail below.

Brief Description of the Drawings

Figure 1 is a floor plan for a DRAM macro according to an embodiment of the invention.

20 Figure 2 is a timing diagram showing the operation of a DRAM macro according to the invention.

Detailed Description of the Invention

The invention provides a synchronous multibank interface on a growable embedded DRAM. The flexible multibank interface of the invention can be grown without redesign and without change of appearance/behavior to the customer. The interface is preferably characterized by the presence of bank select inputs (pins) which permit selection of one or more banks of the macro. The banks preferably each have respective support circuitry including row decode circuitry and respective limited repair redundancy. The interface of the invention has advantages over typical synchronous DRAM interfaces in that operations to each bank are independent. Also, activation and pre-charge operations can both be completely hidden during reads and writes. The interface of the invention may be used to support multiple bank configurations with any number of banks (i.e., one or more banks).

The multi-bank interface of the invention extends the macro performance by allowing concurrent operations to independent banks and thereby allowing improved bus utilization during multi-bank operations. For example, bank activate and precharge commands may be hidden during read and write cycles. Also, the invention allows for multiple activates and precharges to one to N banks where N is the number of banks in the macro.

The invention is illustrated by specific embodiments below. It should be understood that the embodiment is intended to illustrate an example of the invention. The invention encompasses other embodiments which are not described in detail below.

The embodiment described below is a growable eDRAM macro featuring a flexible multi-banking protocol and a column redundancy scheme to support multi-banking. Figure 1 shows a high level floor plan of an eDRAM macro

embodiment according to the invention. Each eDRAM macro 10 preferably contains a bank(s) 20 having an array of DRAM memory elements 25 (preferably, each bank contains a 1Mb array), column redundancy 30, data I/O 40, a control system 50, a common power system 60, buffer 120, data 5 comparator/redundancy allocation 110, a built-in self-test engine (BIST) 70, and supporting ROM/SROM 100. The bank comprising memory array and its support circuitry may be stepped to build the desired macro size. (Figure 1 shows a four bank array.) Thus, a macro of desired size is constructed by replicating the bank the appropriate number of times. If desired, multiple 10 eDRAM macros can be situated on an ASIC chip enabling customers to make the performance/die area trade-off specific to their application.

For a 1Mb bank array, the cells are preferably organized 512 rows by 2048 columns. The columns are preferably divided in groups of 8 X 256, producing a 256 bit data word with an 8 bit page depth. An x292 parity option 15 may be supported by adding 288 columns (36 data bits). This organization provides a high array utilization while maintaining performance typically lost by making longer word-lines or bit-lines. Faster core performance can be achieved with shorter word-lines and bit-lines, but at the cost of more wordline decoder/drivers and/or sense amplifiers. The wide I/O provides maximum 20 bandwidth and, for application not requiring the full width, bit-write control may be included to facilitate masking.

Each bank 20 preferably includes a dedicated redundancy repair region. For a 1Mb bank array, the redundancy repair region preferably contains 8 redundant rows and 8 redundant data bits that can be used to repair faults in that 25 1Mb array. The redundancy repair region is preferably limited whereby it can be incorporated in each bank to enable independent bank operation and facilitate growing the macro.

Each bank 20 is preferably supported by a dedicated row decoder 80 and a shared column decoder and local buffer (data path receiver / driver) 90. Each bank 20 preferably acts as an independent bank sharing a common address and data bus. Each bank 20 preferably shares a local buffer (data path receiver / driver) with the other banks of the macro. Shared column decoder(s) and local buffer are preferably placed adjacent to the sense amplifiers of its respective bank(s). The column decoder preferably acts to reduce the I/O to 8:1. This reduction provides 1 bit of data on approximately the same physical pitch as a 12 track logic cell (4.8 μ m). This feature reduces wiring congestion and facilitates bit slice design internal and external to the DRAM macro. Reducing the width from 2K to 256 provides for enough wiring room to allow independent read and write data buses with ample tracks for power and data shielding.

To support column redundancy with a wide I/O interface (x256), it is preferable to implement redundancy at the data-bit level, as described in US Patent Application Serial No. 09/971840, filed October 5, 2001, the disclosure of which is incorporated herein by reference. The row redundancy circuits compare the current address with stored addresses of defective wordlines. If the incoming addresses match a stored defective address, the defective wordline is held inactive and a spare wordline is activated in its place. Redundancy provides repairability and improved yield.

For multi-banking, the column redundancy preferably switches at the page rate, and therefore sets up in a fraction of the time of conventional column redundancy. For speed, the repair solutions are preferably stored in local latches (one per data-bit per bank) and are rapidly selected by the incoming bank address in time to set shifting switches prior to driving write data to sub-array. A bank's column solution is preferably stored in the local latches as string of '0's and '1's with the transition from '0' to '1' indicating a failing location.

To reduce fuse count, preferably only the failing address is stored. The string is preferably generated by a counter and latch that toggles when counter matches the address of failing locations and loaded via shift chain at power up. The column solution is preferably implemented through column steering switches in
5 column redundancy section 30 and buffer 120.

Each bank select pin (BSN) is associated with a corresponding each bank and is used in conjunction with other command input signals: master select (MSN), Page (PGN), Address (Ax) and Write Enable (WEN) to determine which bank(s) within the macro will respond to input commands. MSN is preferably
10 treated like a master input clock, latching the state of all other input pins with each falling MSN edge as shown in Figure 2.

On the interface to the memory, a bank select input for each bank is brought out as a control pin for that bank. All BSN inputs are defined at every MSN falling edge to indicate whether each bank is to remain open or closed or
15 whether a bank is to become active/open (from the pre-charge state) or become pre-charged/closed (from the active state). The address bus is preferably not multiplexed (i.e., all addresses are available at the same time) to simplify the design. Alternatively, multiplexing can be used if desired (e.g., to support legacy designs, etc.) such that the eDRAM macro may operate similarly to a
20 synchronous DRAM (SDRAM) where addressing is performed in a RAS/CAS type manner, and the master select input (MSN) is treated like a master input clock, latching the state of all other input pins with each falling MSN edge.

Any combination of banks within a macro can be active simultaneously. Preferably, each bank is opened in a sequential fashion. The row address for
25 each active bank will remain latched until the bank is pre-charged. This frees the row address bus to allow other banks to be activated at a different row address

on subsequent MSN clock cycles. No address information is required to pre-charge a bank.

- By latching a BSN in low when it was previously high, an activate is issued. By latching in a BSN high when it was previously low, a precharge is issued. When latching in a BSN low when it was previously low, a write or read to that bank can be issued. When an activate is issued, the row address latched and used to activate that row address to all banks being activated. This interface does not limit the number of BSN pins that may be going low on any one cycle. However, all banks being activated in one cycle will open the same row address so this function may have limited use (refresh is one possibility). The bank address is latched to select which bank has control of the shared data bus.

- Another pin (PGN), paging to memory, is used to enable the write/read operation to an active bank. If PGN is latched in high, no write or read will occur. If PGN is latched in low, the bank specified by the current bank address (BA) will be written or read based on the state of WEN. When a read or write is issued to a bank, the column address is latched and selects which column to write or read from. Table 1 below shows a truth table of the commands.

Table 1: Command Truth Table

	Activate	Read	Write	Refresh	Precharge
BSN	-> 0	0	0	0	-> 1
PGN	X	0	0	1	X
WEN	X	1	0	X	X
Row Address (RA)	V	X	X	V	X
Bank Address (BA)	X	V	V	X	X
Column Address (CA)	X	V	V	X	X

In Table 1, the five basic commands to the eDRAM are shown. A '0' indicates low, '1' indicates high, '-> 0' means going low on that cycle, '-> 1' means going high on that cycle, 'X' indicates a don't care and 'V' indicates a valid address. Note that multiple commands are allowed on a single cycle. For example, an activate to bank 0 and a read to bank 1 can both occur on the same cycle because the state of each pin is either the same or one is a don't care. This allows the customer to completely hide the activate and precharge portions of a memory access during read and writes to other banks therefore optimizing bus utilization. As the eDRAM is grown, the bank count can increase without complicating or changing the interface (other than the number of BSN pins). To handle this, the BSN pins are labeled from BSN<0> to BSN<N> where N+1 is the number of banks. The timing diagram in Figure 2 shows a sample of bank operations to a 4 bank eDRAM. Having individual BSN pins for each bank further enables unambiguous support for a non-binary number of banks (like 3, 5, 6, 7, 9, etc).

In this timing diagram, MSN is shown as an active low clock where signals are latched on the falling edge. BSN0 is the bank select to bank 0, BSN1 is the bank select to bank1, etc. These are active low. PGN is the active low page strobe that activate the data path. WEN is low for a write cycle and high for a read cycle. Note that the row address (A11-A3) is lined up the bank activates where the bank and column addresses (A13-A12 and A2-A0, respectively) are lined up with the write and read operations. DI is the data in bus, DO is the data out bus and BW is a bit mask bus that is used on the eDRAM. On each cycle, a description of the operation is given above MSN. BAx is a bank activate to bank x. PBx is a precharge to bank x. RBx is a read to x, WBx is a write to x. NOP indicates no activate, pre-charge, read or write on that cycle. Note that the precharge of bank1 at 20ns does not interrupt the data bus, and that the activate of bank1 at 25ns to a new row address also does not interrupt the data bus.

[illegible]